Preprint typeset in JHEP style - HYPER VERSION

FERMILAB-PUB-15-122-T

Improved partonic event generators at lepton colliders

Walter T. Giele

Fermilab, P.O.Box 500, Batavia, IL 60510, USA
E-mail: giele@fnal.gov

Abstract: A method is detailed for the phase space integration of $\gamma^* \to$ multi-jets applicable to parton level Monte Carlo's at any order in perturbation theory. Other non-jet objects, massless or massive, can be included in the phase space generation. We correlate the bremsstrahlung events in a manner that integrates out all partonic configurations leading to a fixed jet configuration, thereby improving convergence. This also allows the method to extend infra-red safety to the fully differential multi-jet cross section.

Keywords: QCD.

Contents

1. Introduction

When calculating differential cross sections involving jets the majority of effort and ingenuity is invested in evaluating the matrix elements. The phase space integration and definitions of proper observables are usually afterthoughts. For Leading Order (LO) parton level Monte Carlo's many methods have been proposed to improve the behavior of multi-jet event generation. These methods focus on the properties of the LO amplitude [1–7]. Here we focus on event generation for parton level event generators beyond LO. At higher order partons are clustered, resulting in jets composed of two or more partons. This internal configuration of the jet is not purview to the observer within the context of a fixed order calculation and needs to be averaged, resulting in a perturbative calculable and reliable result.

We will exploit this necessity by constructing a phase space generator which explicitly integrates out all partonic configurations which give rise to the same multi-jet finals state for a particular jet algorithm. In ref. [8] the basic concepts of such an approach to phase space integration was developed and a proof of existence was given. These techniques were refined and modified in ref. [9] for application to the Matrix Element Method (MEM), extending the concept of MEM to Next-to-Leading Order (NLO) [10]. Finally, a first look at how to implement these correlated phase space generators for the calculation of NLO cross sections were highlighted in ref. [11]. In this paper we develop the method fully for parton event generators at lepton colliders, ready to be applied to e.g. fully differential Next-to-Next-to-Leading Order (NNLO) 3-jet cross sections [12].

The standard manner in which phase space event generators work in Monte Carlo's calculating higher order correction to jet cross sections is to generate the events with different parton multiplicities completely uncorrelated. For example, to calculate the NNLO 3-jet cross section at a leptonic collider one first generates many 3-parton events and bins the observable calculated from the 3 momenta, weighted by the value of the sum of the regulated 3-parton amplitudes. Once completed, another run is initiated generating many 4-parton events for which the observable is calculated and binned with the weight of the regulated

4-parton amplitude. Finally, the 5-parton events are generated and the observable binned using the regulated 5-parton amplitude. In this the integration of the observable over the jet phase space plays a crucial role to achieve infra-red safety.

Yet, from the viewpoint of the observed jet final state, these different parton multiplicity events have a high degree of correlation. By applying a jet algorithm, the hadronic final state is simplified into a final state of jet objects, each jet having a momentum. Perturbative QCD should be able to calculate these jet cross sections, i.e. the correlations between the jet momenta, order by order in perturbation theory provided the jets are opaque. That is, the jets are averaged over all hadronic configurations leading to that particular jet configuration. As a consequence the fully differential jet cross section should be calculable given an appropriate infra-red safe jet algorithm. For the NNLO 3-jet example this will significantly alter the manner in which the 3-parton, 4-parton and 5-parton final states are generated. First of all, the generator calculates the fully differential 3-jet cross section $d\sigma_3/dp_1dp_2dp_3$ where $\{p_i\}$ are the jet axis momenta. Observables can be calculated by integrating the jet observable over the jet phase space. Note that nowhere it is required to average over the jet phase space to obtain an infra-red safe answer, nor are there any constraints on the type of jet observable one can look at. For the event generator this means that the starting point is a given fixed 3-jet configuration (i.e. the jet momenta $\{p_i\}$ are given). This requires only a single evaluation of the 3-parton amplitudes. From this starting point we generate 4-parton events which reconstruct back to the initiating 3-jet momenta using the jet algorithm. Because the jets are fixed the 4-parton amplitude weights can simply be added to the 3-parton amplitude weight. Due to the recursive nature of sequential clustering in jet algorithm we can repeat the above argument for the 5-parton contribution: given a single 4-parton event we can generate many 5-parton events such that applying a single step in the jet algorithm leads back to the initiating 4-parton event. It is clear that this will lead to a highly correlated event generator and the event weight of the fully differential 3-jet cross section is a series expansion in the strong coupling constant at the scale of the jet resolution.

Current jet algorithms do not extend the concept of infra-red safety to the fully differential jet cross sections. This is solely due to the clustering phase in the algorithm, i.e. how to combine two momenta to form a new one. The new momentum formed from the momenta of the respective clustered particles is simply the sum of these momenta. For some algorithms this new momentum is redefined to make it a massless momentum (see ref. [13] for an overview of these schemes). As a result applying the jet algorithm on e.g. a 4-parton final state to form a 3-jet final state will never overlap with the 3-jet final state generated from the 3-parton final state due to jet mass and/or non-momentum conservation.

As a consequence we have to adjust the cluster phase of the jet algorithms in order to obtain infra-red safe fully differential jet cross sections. Note that one can still apply any ordinary jet algorithm using the above described correlated phase space generator. However the now theoretical jet algorithm used internally by the generator to reorganize the correlated phase space generation does not match the applied jet algorithm exactly. The price to pay is that one must define observables and average this jet observable by

integrating over the full jet phase space so that an infra-red safe prediction can be made. Still, the generation of the correlated multi-parton final states could greatly benefit the convergence of the predictions of the observable made by the Monte Carlo.

In section 2 we will construct the required forward brancher which converts an n-particle phase space into a $(n+1)$-particle phase space, invertible using the cluster algorithm. In section 3 this Forward Branching Phase Space (FBPS) generator is used to construct the correlated event generators needed for calculating fully differential multi-jet cross sections at any order in perturbation theory. We conclude in section 3 by summarizing the results and outline the next steps in further developing the FBPS method.

2. Constructing the forward branching phase space generator

The first step in the construction of the FBPS is to define a proper clustering algorithm. As already explained in the introduction, we need a clustering algorithm which maps a massless n-particle phase space onto a massless $(n-1)$-particles phase. [1]

Current jet algorithms simply combine two momenta by adding the 4-vectors. Take as an example the decay of a heavy particle with momentum Q into four massless particles, clustering momentum p_3 and momentum p_4

$$Q = \hat{p}_1 + \hat{p}_2 + \hat{p}_3 + \hat{p}_4 = \hat{p}_1 + \hat{p}_2 + \hat{p}_{34} \,. \tag{2.1}$$

This obviously lead to a 3-particle phase space where one of the momenta is massive. Some clustering algorithms rescale the energy or momentum such that the clustered particle is massless (see ref. [13]). However such a scheme would change Q through momentum conservation.

The way to modify the clustering is to use a recoil momentum. By rescaling at least one of the other non-clustered particles one can make the clustered particle massless (this is in essence a $3 \to 2$ clustering as is commonly used to construct the subtraction terms needed to regulate various amplitudes in higher order calculations [14]). Explicitly, eq. 2.1 becomes

$$Q = \hat{p}_1+\hat{p}_2+\hat{p}_3+\hat{p}_4 = \hat{p}_1+\hat{p}_2+\hat{p}_{34} = (1-\beta)(\hat{p}_1+\hat{p}_2)+(\hat{p}_{34}+\beta(\hat{p}_1+\hat{p}_2)) = p_1+p_2+p_3 \,, \tag{2.2}$$

where we have used the sum of all other particles as a recoil momentum so to avoid further complicating the algorithm on how to select a recoil particle. The new momenta are given by $p_1 = (1-\beta)\hat{p}_1$, $p_2 = (1-\beta)\hat{p}_2$ and $p_3 = \hat{p}_{34} + \beta(\hat{p}_1 + \hat{p}_2)$. The branching scale variable β can now be chosen such that the clustered particle is massless

$$\begin{aligned} p_3^2 &= (\hat{p}_{34} + \beta\hat{p}_{12})^2 = \beta^2 \hat{s}_{12} + 2\beta\, \hat{p}_{34} \cdot p_{12} + \hat{p}_{34}^2 = 0 \\ \Rightarrow \beta &= \frac{(\hat{p}_{34} \cdot \hat{p}_{12}) - \sqrt{(\hat{p}_{34} \cdot \hat{p}_{12})^2 - \hat{s}_{12}\hat{p}_{34}^2}}{\hat{s}_{12}} \,, \end{aligned} \tag{2.3}$$

[1] Note that it is straightforward to add other, massive or massless, momenta as long as they to not participate in the jet clustering. For the remainder of the paper we ignore this in order to simplify the notation and discussion.

were $p_{ij} = p_i + p_j$ and $s_{ij} = p_{ij}^2$. With this simple augmentation of the cluster algorithm we have what we need to construct the cluster invertible FBPS generators.

It is convenient to introduce here the inverse of the clustering kinematics, i.e. the branching kinematics. Introducing the branching scaling variable α we have

$$Q = p_1+p_2+p_3 = (1+\alpha)(p_1+p_2)+(p_3-\alpha(p_1+p_2)) = \hat{p}_1+\hat{p}_2+\hat{p}_{34} = \hat{p}_1+\hat{p}_2+\hat{p}_3+\hat{p}_4 \ , \quad (2.4)$$

where α is given by the quadratic equation

$$\hat{s}_{34} = \hat{p}_{34}^2 = (p_3 - \alpha(p_1 + p_2))^2 = \alpha^2 s_{12} - 2\alpha\, p_3 \cdot (p_1 + p_2) \ . \quad (2.5)$$

The branching and merging scale variables are related by

$$1 - \beta = \frac{1}{1+\alpha} \Rightarrow \beta = \frac{\alpha}{1+\alpha} \ , \quad (2.6)$$

and give the relations to be used later

$$\beta(\hat{p}_1 + \hat{p}_2) = \beta\hat{p}_{12} = \beta(1+\alpha)p_{12} = \alpha p_{12} = \alpha(p_1 + p_2); \ d\beta = \frac{d\alpha}{(1+\alpha)^2} \ . \quad (2.7)$$

Having augmented the clustering algorithm we can now start making the FBPS generator needed to generate the massless $(n+1)$-parton phase space from the massless n-parton phase space subject to the clustering constraint of eq. 2.2.

The n-particle fully differential cross section is given by

$$\frac{d\sigma}{dp_1 \cdots dp_n} = \left(\frac{(2\pi)^4}{2\sqrt{Q^2}}\right) \times d\,\Phi_n(Q; p_1, \ldots, p_n) \ . \quad (2.8)$$

The flat phase space is given by

$$d\,\Phi_n(Q; p_1, \ldots, p_n) = \prod_{i=1}^{n} \frac{d^4 p_i}{(2\pi)^3} \delta^+(p_i^2)\, \delta^4(Q - p_1 - \cdots - p_n) \ . \quad (2.9)$$

We will derive the FBPS generating a massless 4-particle phase space from a massless 3-particle phase space. Afterward we will generalize this to n-particle phase spaces. The first step in constructing the FBPS generator is simply generating the massless 4-particle phase space from a 3-particle phase space with one massive particle

$$d\,\Phi_4(Q; \hat{p}_1\hat{p}_2\hat{p}_3\hat{p}_4) = (2\pi)^3 d\hat{s}_{34}\ \Phi_3(Q; \hat{p}_1\hat{p}_2\hat{p}_{34}) \times \Phi_2(\hat{p}_{34}; \hat{p}_1\hat{p}_2) \ , \quad (2.10)$$

where $\hat{p}_{34}^2 = \hat{s}_{34}$.

The next step is to implement the rescaling of momentum $\hat{p}_{34}$ using the recoil momentum $\hat{p}_{12}$. Specifically, starting from the massive phase space

$$d\Phi_3(Q; \hat{p}_1\hat{p}_2\hat{p}_{34}) = \frac{d^4\hat{p}_1}{(2\pi)^3}\frac{d^4\hat{p}_2}{(2\pi)^3}\frac{d^4\hat{p}_{34}}{(2\pi)^3}\delta^+(\hat{p}_1^2)\delta^+(\hat{p}_2^2)\delta^+(\hat{p}_{34}^2-\hat{s}_{34})\delta^4(Q-\hat{p}_1-\hat{p}_2-\hat{p}_{34}) \ , \quad (2.11)$$

we have to derive the Jacobean J generated by the change of the integration momenta

$$
\begin{aligned}
d\Phi_3(Q;\hat{p}_1\hat{p}_2\hat{p}_{34}) &= J\times d\Phi_3(Q;p_1p_2p_3) \\
&= J\times\frac{d^4p_1}{(2\pi)^3}\frac{d^4p_2}{(2\pi)^3}\frac{d^4p_3}{(2\pi)^3}\delta^+(p_1^2)\delta^+(p_2^2)\delta^+(p_3^2)\delta^4(Q-p_1-p_2-p_3)\ .
\end{aligned}
\tag{2.12}
$$

To calculate the Jacobian J we need to rewrite the integrals over the momenta $\hat{p}_1$,$\hat{p}_2$ and $\hat{p}_{34}$ into integrals over momenta p_1, p_2 and p_3. To do this we mathematically express the clustering algorithm as a decomposition of unity

$$
\begin{aligned}
1 &= \hat{s}_{12}(\beta_+-\beta_-)\int d\beta\, dp_1\, dp_2\, dp_3\delta((1-\beta)\hat{p}_1-p_1)\delta((1-\beta)\hat{p}_2-p_2)\delta(p_3^2) \\
&\times\ \delta(p_3-(\hat{p}_{34}+\beta\hat{p}_{12}))\ ,
\end{aligned}
\tag{2.13}
$$

where $\beta_\pm$ are given by solving equation $(\hat{p}_{34}+\beta\hat{p}_{12})^2=0$:

$$
\hat{s}_{12}(\beta_+-\beta_-)=2\sqrt{(\hat{p}_{34}\cdot\hat{p}_{12})^2-\hat{s}_{34}\hat{s}_{12}}\ . \tag{2.14}
$$

The above identity is expressed in terms of the cluster scale variable β and cluster kinematics. We rewrite this equation in terms of the branching scale variable α and branching kinematics, giving us

$$
\begin{aligned}
1 &= 2\int d\alpha\, dp_1\, dp_2\, dp_3(1+\alpha)^7\sqrt{(\hat{p}_{34}\cdot p_{12})^2-\hat{s}_{34}s_{12}} \\
&\times\ \delta(\hat{p}_1-(1+\alpha)p_1)\delta(\hat{p}_2-(1+\alpha)p_2)\delta(p_3^2)\delta(\hat{p}_{34}-(p_3-\alpha\, p_{12}))\ .
\end{aligned}
\tag{2.15}
$$

Multiplying this decomposition of one to eq. 2.11 and integrating the momenta $\hat{p}_1$, $\hat{p}_2$ and $\hat{p}_{34}$ over the appropriate δ-functions one obtains

$$
\begin{aligned}
d\Phi_3(Q;\hat{p}_1\hat{p}_2\hat{p}_{34}) &= \int d\alpha(1+\alpha)^3\sqrt{(\hat{p}_{34}\cdot p_{12})^2-\hat{s}_{34}s_{12}}\ \delta(\alpha^2 s_{12}-2\alpha\, p_3\cdot p_{12}-\hat{s}_{34}) \\
&\times\left[\frac{dp_1}{(2\pi)^3}\frac{dp_2}{(2\pi)^3}\frac{dp_3}{(2\pi)^3}\delta(p_1^2)\delta(p_2^2)\delta(p_3^2)\delta(Q-p_1-p_2-p_3)\right]\ ,
\end{aligned}
\tag{2.16}
$$

with $\hat{p}_1=(1+\alpha)p_1$, $\hat{p}_2=(1+\alpha)p_2$ and $\hat{p}_{34}=p_3-\alpha\, p_{12}$.

By integrating over the branching variable α and selecting the physical solution of the quadratic equation we obtain the final expression

$$
d\Phi_3(Q;\hat{p}_1\hat{p}_2\hat{p}_{34})=(1+\alpha_-)^3\sqrt{\frac{(\hat{p}_{34}\cdot p_{12})^2-\hat{s}_{34}s_{12}}{(\hat{p}_{34}\cdot p_{12})^2+\hat{s}_{34}s_{12}}}\times d\Phi_3(Q;p_1p_2p_3)\ , \tag{2.17}
$$

with

$$
\alpha_-=\frac{\hat{p}_{34}\cdot p_{12}-\sqrt{(\hat{p}_{34}\cdot p_{12})^2+\hat{s}_{34}s_{12}}}{s_{12}};\ \hat{p}_1=(1+\alpha_-)p_1;\ \hat{p}_2=(1+\alpha_-)p_2;\ \hat{p}_{34}=p_3-\alpha_-\, p_{12}\ . \tag{2.18}
$$

It is now straightforward to generalize this result. We can generate the massless n-particle phase space from a massless $(n-1)$-particle phase space using the FBPS generator. Specifically, by branching massless particle j we obtain

$$
d\,\Phi_n(Q;\{\hat{p}\})=d\,\Phi_{n-1}(Q;\{p\})\times d\,\Phi_{\rm fbps}^{[j]}(\{\hat{p}\}|\{p\})\ , \tag{2.19}
$$

where[2]

$$
\begin{aligned}
d\,\Phi^{[j]}_{\text{fbps}}(\{\hat{p}\}|\{p\}) &= (2\pi)^3\,(1+\alpha_j)^{2n-3} \times \sqrt{\frac{(\hat{p}_{jn}\cdot Q_j)^2 - \hat{s}_{jn}Q_j^2}{(p_j\cdot Q_j)^2 + \hat{s}_{jn}Q_j^2}}\, d\hat{s}_{jn}\, d\,\Phi_2(p_{jn};\hat{p}_j\hat{p}_n) \\
\hat{p}_i &= (1+\alpha_j)\,p_i \quad (i \neq j) \\
\hat{p}_{jn} &= p_j - \alpha_j Q_j = (1+\alpha_j)p_j - \alpha_j Q \\
Q_j &= \sum_{i\neq j} p_i = Q - p_j \\
\alpha_j &= \frac{(p_j\cdot Q_j) - \sqrt{(p_j\cdot Q_j)^2 + \hat{s}_{jn}Q_j^2}}{Q_j^2}\,.
\end{aligned}
\tag{2.20}
$$

We introduced the notation $\{p\} = p_1, p_2, \ldots$ for a list of momenta in order to simplify the notation. The number of momenta in the list is clear from the context.

By iterating the above FBPS, we can generate a n-particle phase space from a m-particle phase space (with at least 1 massless particle) through intermediary k-particle phase spaces. Furthermore, we can invert the generated n-particle phase space by applying the appropriate clusterings as defined in eq. 2.2.

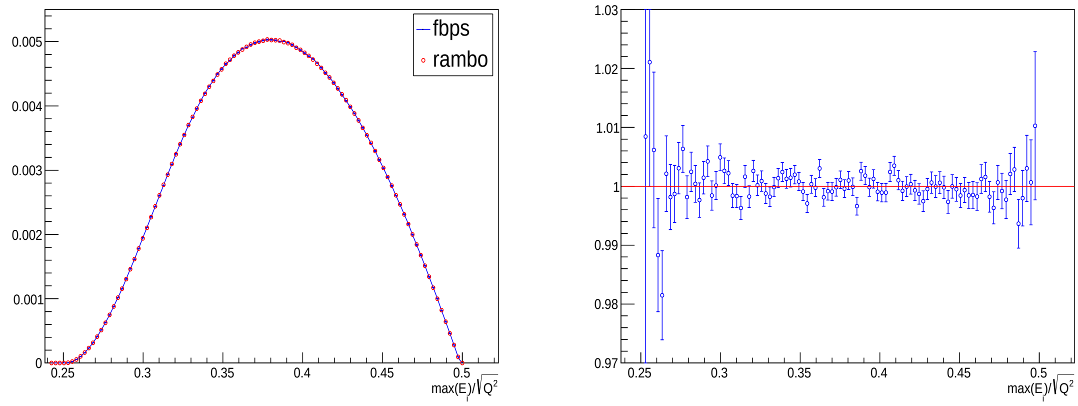

Figure 1: Comparison of the flat phase space distribution and the $3 \to 4$ FBPS generated distribution for the observable $\max_i(E_i)/\sqrt{Q^2}$ using 10^8 events. The figure on the right is the ratio of the two distributions. Both the 3-particle and 4-particle flat phase space were generated using RAMBO and $\sqrt{Q^2} = 240$. The results in the left graph were not rescaled for the bin-width.

Finally we want to validate the correctness of eq. 2.19. This can be done using a Monte Carlo integration of the FBPS generated results versus the results obtained using the flat phase space generator RAMBO [15].

The first check is to see whether the FBPS generated phase space has the correct phase

[2] If we start from the special configuration $d\,\Phi(Q, p_1p_2)$ where both momenta are massless, we get the solution $\alpha = -\hat{s}_{23}/s_{12} = -\hat{s}_{23}/Q^2$ when branching momentum p_2.

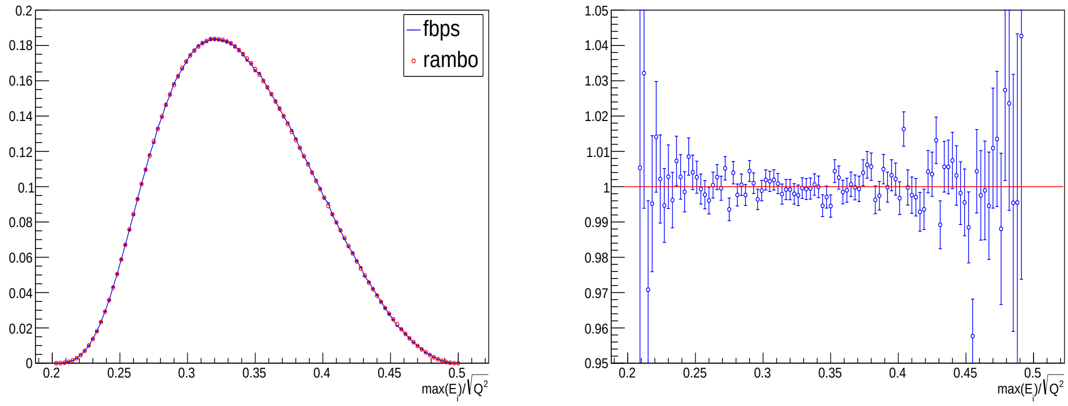

Figure 2: Comparison of the flat phase space distribution generated by and the $3 \to 4 \to 5$ generated FBPS distribution for the observable $\max_i(E_i)/\sqrt{Q^2}$ using 10^8 events. The figure on the right is the ratio of the two distributions. Both the 3-particle and 5-particle flat phase space were generated using RAMBO and $\sqrt{Q^2} = 240$. The results in the left graph were not rescaled for the bin-width.

space volume. The volume of phase space is given by

$$\int d\,\Phi_n(Q;\{p^{(n)}\}) = \int d\,\Phi_m(Q;\{p^{(m)}\}) \times \left[\prod_{k=1}^{n-m} d\,\Phi^{[j_k]}_{\text{fbps}}(\{p^{(m+k)}\}|\{p^{(m+k-1)}\})\right]$$
$$= \left(\frac{\pi}{2}\right)^{n-1} \frac{1}{(2\pi)^{3n}} \frac{(Q^2)^{n-2}}{(n-1)!(n-2)!}\,. \tag{2.21}$$

We numerically verified the thus generated n-particle phase space, for all $3 \leq m < n \leq 10$, gives back the correct phase space volume. For each event generated in the Monte Carlo all j_k were chosen randomly. The m-particle phase space in the comparison was generated using the flat phase space generator RAMBO.

To further validate the FBPS generator we made the differential cross section for the observable $\max_i(E_i)/\sqrt{Q^2}$ where E_i is the energy of momentum p_i for both the $3 \to 4$ FBPS generator in figure 1 and the $3 \to 5$ FBPS generator in figure 2. The results were compared against the predictions from RAMBO for the 4- and 5-particle phase space respectively. As can be seen in both cases the agreement is as expected, given the 10^8 generated events used in the Monte Carlo's.

3. Constructing differential jet cross sections

Now that we have the basic building block in the form of the FBPS generator as detailed in eq. 2.19, we can construct the event generator to calculate the fully differential jet cross sections. The event generator will depend on the resolution function $d_{ij} = d(p_i, p_j)$ of the jet algorithm. The pair of momenta with the smallest d_{ij} will be clustered by the jet algorithm and it will keep clustering pairs of momenta until it reaches the condition $\min_{ij}(d_{ij}) > d_{\text{cut}}$ (exclusive jet cross section) or until a certain multiplicity of momenta is

reached (i.e.. keep clustering until we reach m momenta, giving us an inclusive m-jet cross section).

We will start with eq. 2.19 and extend it to give us the generator to calculate the fully differential jet cross sections. The jet resolution function d_{ij} will do two things for us. It will give which momentum to branch and set an upper integration boundary on the variable $\hat{s}_{jn}$ and decay products of $\hat{p}_{jn}$. We can implement the branches by using the resolution function to partition the $(n-1)$-particle phase space into wedges. Each wedge is associated with a momentum to branch in order to generate the n-particle phase space. To do this we define the following partition of one

$$1 = \sum_{j=1}^{n-1} \theta(\hat{d}_{jn} = \hat{d}_{\min}) \ , \tag{3.1}$$

where the θ-function equals one if its argument is true and zero otherwise. The argument of the θ-function $\hat{d}_{\min} = \min_{ij}(\hat{d}_{ij}) = \min_{ij}(d(\hat{p}_i, \hat{p}_j))$ is true as long as $\hat{d}_{jn}$ has the smallest resolution parameter (i.e. it will be the pair which will be clustered by the jet algorithm). This partition divides phase space in the desired wedges. In wedge j, particle p_j branches to give particles $\{\hat{p}_j, \hat{p}_n\}$ in such a manner that when applying the jet algorithm it will pick this pair to cluster back to p_j as $\hat{d}_{jn}$ has the smallest resolution parameter. Note that this partitioning include n-jet final states if we demand a clustering cutoff when $d_{\min} > d_{\text{cut}}$. This distinction can be implemented as

$$1 = \sum_{j=1}^{n-1} \theta(\hat{d}_{jn} = \hat{d}_{\min}) \Big(\theta(\hat{d}_{\min} < \hat{d}_{\text{cut}}) + \theta(\hat{d}_{\min} > \hat{d}_{\text{cut}}) \Big) \ . \tag{3.2}$$

The $(n-1)$-particle jet exclusive FBPS generator is now given by multiplying eq. 2.19 with the above partition of one

$$d\,\Phi_n(Q; \{\hat{p}\}) = d\,\Phi_{n-1}(Q; \{p\}) \times d\,\Phi^{\text{excl}}_{\text{fbps}}(\{\hat{p}\}|\{p\}) \tag{3.3}$$

with

$$d\,\Phi^{\text{excl}}_{\text{fbps}}(\{\hat{p}\}|\{p\}) = \sum_{j=1}^{n-1} \theta(\hat{d}_{jn} = \hat{d}_{\min}) \theta(\hat{d}_{\min} < \hat{d}_{\text{cut}}) d\,\Phi^{[j]}_{\text{fbps}}(\{\hat{p}\}|\{p\}) \ . \tag{3.4}$$

This FBPS generator will produce all bremsstrahlung radiation momenta for a fixed $(n-1)$-jet configuration with jet momenta $\{p_i\}$ provided the modified clustering of eq. 2.2 is used. By integrating over the jet phase space and removing the $\hat{d}_{\text{cut}}$-constraint, the full n-particle phase space is obtained. We can iteratively apply this FBPS generator to obtain the multiple bremsstrahlung phase spaces.

To define the fully differential jet cross section at NNLO using the improved clustering is now straightforward. Given the n-jet configuration with massless jet momenta $\{p\}$ we get

$$\frac{d^{3n}\sigma^{\text{NNLO}}_{\text{impr}}}{d^3p_1 \cdots d^3p_n} = \mathcal{A}(\{p\}) + \int d\,\Phi^{\text{excl}}_{\text{fbps}}(\{\hat{p}\}|\{p\}) \times \left[\mathcal{B}(\{\hat{p}\}) + \int d\,\Phi^{\text{excl}}_{\text{fbps}}(\{\hat{\hat{p}}\}|\{\hat{p}\})\, \mathcal{C}(\{\hat{\hat{p}}\}) \right] \ , \tag{3.5}$$

where

$$\mathcal{A}(\{p\}) = \mathcal{A}(p_1\cdots p_n) = \Big|\mathcal{M}^{(0)}(p_1\cdots p_n)+\mathcal{M}^{(1)}(p_1\cdots p_n)+\mathcal{M}^{(2)}(p_1\cdots p_n)\Big|^2$$
$$\mathcal{B}(\{\hat{p}\}) = \mathcal{B}(\hat{p}_1\cdots \hat{p}_n) = \Big|\mathcal{M}^{(0)}(\hat{p}_1\cdots \hat{p}_n)+\mathcal{M}^{(1)}(\hat{p}_1\cdots \hat{p}_n)\Big|^2$$
$$\mathcal{C}(\{\hat{\hat{p}}\}) = \mathcal{C}(\hat{\hat{p}}_1\cdots \hat{\hat{p}}_n) = \Big|\mathcal{M}^{(0)}(\hat{\hat{p}}_1\cdots \hat{\hat{p}}_n)\Big|^2 \ , \tag{3.6}$$

are the appropriate set of matrix elements at each parton multiplicty. Note that $d\,\Phi_{\rm fbps}^{\rm excl}$ is a 3-dimensional integral, so that the above integral is at most 6-dimensional.

We can still use the FBPS generator and apply other jet algorithms to its produced events. There should still be an advantage over the usual uncorrelated generation of the different phase spaces. Given the jet algorithm mapping $\Delta(\{p\}|\{\hat{p}\})$ the differential cross section is given by

$$\frac{d^{3n}\sigma_\Delta^{\rm NNLO}}{d^3p_1\cdots d^3p_n} = \mathcal{A}(\{p\})\Delta(\{p\}|\{p\})$$
$$+\int d\,\Phi_{\rm fbps}^{\rm excl}(\{\hat{p}\}|\{p\})\Delta(\{p\}|\{\hat{p}\})\times\left[\mathcal{B}(\{\hat{p}\})+\int d\,\Phi_{\rm fbps}^{\rm excl}(\{\hat{\hat{p}}\}|\{\hat{p}\})\Delta(\{\hat{p}\}|\{\hat{\hat{p}}\})\,\mathcal{C}(\{\hat{\hat{p}}\})\right] \ . \tag{3.7}$$

However this fully differential cross section is ill-defined and not infra-red safe. Suppose we choose all the jet momenta massless, then the bremsstrahlung events will never contribute because the bremsstrahlung events generate either jets with masses or momentum Q is changed. That is, in this case the virtual and bremsstrahlung events do not merge. To obtain infrared safety in this case we must define an appropriate observable such that the bremsstrahlung and virtual contributions are sufficiently merged, that is

$$\frac{d\,\sigma_\Delta^{\rm NNLO}}{d\,\mathcal{O}} = \int d\,\Phi(Q;\{p\})\,\delta(\mathcal{O}-\mathcal{O}(\{p\}))\frac{d^{3n}\sigma_\Delta^{\rm NNLO}}{d^3p_1\cdots d^3p_n} \ . \tag{3.8}$$

In this sense the improved jet algorithm is superior to the standard jet algorithms as it extends infra-red safety to the fully exclusive jet cross section.

We can generalize eq. 3.5 and 3.7 using a generating functional Γ

$$\frac{d^{3n}\sigma_{\rm impr}}{d^3p_1\cdots d^3p_n} = \mathcal{A}(\{p\})\times\Gamma_{\rm impr}(\{p\});\ \ \frac{d^{3n}\sigma_\Delta}{d^3p_1\cdots d^3p_n} = \mathcal{A}(\{p\})\times\Gamma_\Delta(\{p\}) \ , \tag{3.9}$$

to generate the n-th order fully differential cross section calculator by expanding n times the Γ-functional

$$\Gamma_{\rm impr}(\{p\}) = 1+\int d\,\Phi_{\rm fbps}^{\rm excl}(\{\hat{p}\}|\{p\})\times\left(\frac{\mathcal{A}(\{\hat{p}\})}{\mathcal{A}(\{p\})}\right)$$
$$\Gamma_\Delta(\{p\}) = 1+\int d\,\Phi_{\rm fbps}^{\rm excl}(\{\hat{p}\}|\{p\})\times\Delta(\{\hat{p}\}|\{p\})\times\left(\frac{\mathcal{A}(\{\hat{p}\})}{\mathcal{A}(\{p\})}\right) \ . \tag{3.10}$$

These generators can readily be implemented into Monte Carlo programs.

4. Conclusions

We constructed a correlated bremsstrahlung parton level event generator which should lead to faster convergence of Monte Carlo phase space integrations for higher order jet cross sections at lepton colliders. The bremsstrahlung events are generated highly correlated, enabling better cancellations between real and virtual contributions. One can use any jet algorithm on the generated events, however using the augmented jet algorithm enables us to define the fully exclusive multi-jet differential cross section. That is, for any given jet configuration the generator will integrate out all radiation inside the jets, thereby making them opaque, and combine the matrix element weights of the different multiplicities unencumbered by any constraint. In other words we can define a probability density (or K-factor), calculable order by order in perturbation theory, for every exclusive jet configuration.

The next step is to extend this method to hadron colliders and apply the above method to NLO and NNLO multi-jet generators at lepton and hadron colliders.

Acknowledgments

We acknowledge useful discussions with Keith Ellis and John Campbell. This research is supported by the US DOE under contract DE-AC02-07CH11359.

References

[1] R. Kleiss and R. Pittau, Comput. Phys. Commun. **83**, 141 (1994) [hep-ph/9405257].

[2] P. D. Draggiotis, A. van Hameren and R. Kleiss, Phys. Lett. B **483**, 124 (2000) [hep-ph/0004047].

[3] C. G. Papadopoulos, Comput. Phys. Commun. **137**, 247 (2001) [hep-ph/0007335].

[4] A. van Hameren and R. Kleiss, Eur. Phys. J. C **17**, 611 (2000) [hep-ph/0008068].

[5] A. van Hameren, arXiv:1003.4953 [hep-ph].

[6] A. van Hameren and C. G. Papadopoulos, Eur. Phys. J. C **25**, 563 (2002) [hep-ph/0204055].

[7] T. Gleisberg and S. Hoeche, JHEP **0812**, 039 (2008) [arXiv:0808.3674 [hep-ph]].

[8] W. T. Giele, G. C. Stavenga and J. C. Winter, arXiv:1106.5045 [hep-ph].

[9] J. M. Campbell, W. T. Giele and C. Williams, JHEP **1211**, 043 (2012) [arXiv:1204.4424 [hep-ph]].

[10] J. M. Campbell, R. K. Ellis, W. T. Giele and C. Williams, Phys. Rev. D **87**, no. 7, 073005 (2013) [arXiv:1301.7086 [hep-ph]].

[11] C. Williams, J. M. Campbell and W. T. Giele, PoS RADCOR **2013**, 037 (2013) [arXiv:1311.5811 [hep-ph]].

[12] A. Gehrmann-De Ridder, T. Gehrmann, E. W. N. Glover and G. Heinrich, Comput. Phys. Commun. **185**, 3331 (2014) [arXiv:1402.4140 [hep-ph]].

[13] S. Bethke, Z. Kunszt, D. E. Soper and W. J. Stirling, Nucl. Phys. B **370**, 310 (1992) [Erratum-ibid. B **523**, 681 (1998)].

[14] S. Catani and M. H. Seymour, Nucl. Phys. B **485**, 291 (1997) [Erratum-ibid. B **510**, 503 (1998)] [hep-ph/9605323].

[15] R. Kleiss, W. J. Stirling and S. D. Ellis, Comput. Phys. Commun. **40**, 359 (1986).